Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system, the distributed testing system including a main server component system and at least two client component systems, comprising:

generating at least one data object <u>as a result of testing the software product, the data object</u> to be locally stored in a first location of a first memory of the first client component system;

registering the first location with the main server component system [[and]]

<u>requesting the data object from the first memory of the first client component system</u> <u>for the second client component system through the main server component, the second client</u> <u>component requesting the data object if the data object is needed to continue further testing of</u> <u>the software product using the second client component system;</u>

transferring the data object from the first memory of the first client component system to a second memory of the second client component system; ~~the transferring being in response to the second client component system requesting the data object from the first client component system through the main server component system~~ <u>and</u>

<u>using the data object to continue testing of the software product on the second client</u> <u>component system</u>.

2. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 1, wherein the registering the first

Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

location further comprising, storing the first location in a shared object table of the main

server component system.

3. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 1, further comprising, registering

the first location with the first client component system.

4. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 3, wherein the registering the first

location further comprising, storing the first location in a client table of the first client

component system.

5. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 1, wherein the registering the first

location with the main server component system is defined by one of a put function and a

putb function.

6. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 1, wherein the data object is defined

by one of a resulting test data generated after executing a portion of a test at the first client

component system, an identification key, a dynamically generated Java class, a configuration

file, a property file, and an initial test data transmitted by the main server component system

to each of the first and second client component systems to initialize the test.

7. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 6, wherein the identification key is

Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

used in a secured protocol to establish a secured communication between the main server component system and each of the first and second client component systems.

8. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 1, wherein the each of the first and second memory is defined by one of a random-access memory (RAM), a dynamic RAM (DRAM), and a static RAM (SRAM).

9. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 1, wherein the first location is a memory address of the first memory.

10. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 1, wherein the second client component system requesting the data object from the first client component system is defined by one of a get function and a getb function.

11. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 1, further comprising,

executing a distributed test harness on the main server component system; and

executing a client harness on each of the first and second client component systems.

12. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system, the distributed testing system including a main server component system and at least two client component systems, comprising:

Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

executing a portion of a <u>software product</u> test at a first client component system, the

executing generating at least one data object <u>associated with the software product test</u> to be

locally stored in a first location of a first memory of the first client component system;

registering the first location with the main server component system; [[and]]

transferring the data object from the first memory of the first client component

system to a second memory of the second client component system; the transferring being in

response to the second client component system requesting the data object from the first

client component system through the main server component system<u>; and</u>

<u>executing another portion of the software product test at the second client component</u>

<u>system using the data object from the first client component system</u>.

13. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 12, wherein the registering the first

location further comprising,

storing the first location in a shared object table of the main server component

system.

14. (Currently amended) A <u>computer-implemented</u> method for testing a software

product in a distributed testing system as recited in claim 12, further comprising,

registering the first location with the first client component system.

Appl. Nb. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

15. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 14, wherein the registering the first location further comprising,

storing the first location in a client table of the first client component system.

16. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 12, wherein the registering the first location with the main server component system is defined by one of a put function and a putb function.

17. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 12, wherein the each of the first and second memory is defined by one of a random-access memory (RAM), a dynamic RAM (DRAM), and a static RAM (SRAM).

18. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 12, wherein the first location is a memory address of the first memory.

19. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 12, wherein the second client component system requesting the data object from the first client component system is defined by one of a get function and a getb function.

Appl. Nó. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

20. (Currently amended) A <u>computer-implemented</u> method for testing a software product in a distributed testing system as recited in claim 12, further comprising,

executing a distributed test harness on the main server component system; and

executing a client harness on each of the first and second client component systems.

21. (Currently amended) A system for testing a software product in a distributed testing system, comprising:

a main server component system <u>that manages testing of the software product on client component systems</u>;

a first client component system, the first client component system being in communication with the main server component system <u>to test the software product</u>;

a data object located in a memory of the first client component system<u>, the data object a result of the software product test conducted by the first client component system</u>;

a second client component system, the second client component system being in communication with the main server component system <u>to test the software product</u>; ~~and~~

wherein the main server component system facilitates communication between each of the first and second client component systems, and wherein a location of the data object is registered with the main server component system for use in transferring the data object from the first client component system to the second client component system <u>if the second client component needs the data object to further test the software product</u>.

Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

22. (Original) A system for testing a software product in a distributed testing system as recited in claim 21, further comprising,

a distributed test harness executed on the main server component system;

a first client harness executed on the first client component system; and

a second client harness executed on the second client component system.

23. (Original) A system for testing a software product in a distributed testing system as recited in claim 21, wherein the main server component system includes a shared object table for storing the location of the data object.

24. (Original) A system for testing a software product in a distributed testing system as recited in claim 21, wherein the first client component system includes a client table for storing the location of the data object.

25. (Original) A system for testing a software product in a distributed testing system as recited in claim 21, further comprising a data transfer monitor facility coupled to the main server component system.

26. (Original) A system for testing a software product in a distributed testing system as recited in claim 21, wherein the data object is defined by one of a resulting test data generated after executing a portion of a test at the first client component system, an identification key, a dynamically generated Java class, a configuration file, a property file,

Appl. No. 10/637,132
Amdt. dated February 16, 2007
Reply to Office action of October 16, 2006

PATENT

and an initial test data transmitted by the main server component system to each of the first

and second client component systems to initialize the test.

27. (Original) A system for testing a software product in a distributed testing system

as recited in claim 26, wherein the identification key is used in a secured protocol to

establish a secured communication between the main server component system and each of

the first and second client component systems.

28. (Original) A system for testing a software product in a distributed testing system

as recited in claim 21, wherein the memory is defined by one of a random-access memory

(RAM), a dynamic RAM (DRAM), and a static RAM (SRAM).

29. (Original) A system for testing a software product in a distributed testing system

as recited in claim 21, wherein the location is a memory address of the memory.